S29-98-041

Europäisches Patentamt | European Patent Office | Office européen des brevets

Bescheinigung | Certificate | Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. | Patent application No. | Demande de brevet n°

99101966.2

Der Präsident des Europäischen Patentamts:
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

Arlette Fiedler

A. Fiedler

DEN HAAG, DEN
THE HAGUE,
LA HAYE, LE 04/08/99

Europäisches Patentamt
European Patent Office
Office européen des brevets

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:
Demande n°: 99101966.2

Anmeldetag:
Date of filing:
Date de dépôt: 01/02/99

Anmelder:
Applicant(s):
Demandeur(s):
International Business Machines Corporation
Armonk, NY 10504
UNITED STATES OF AMERICA

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
Personal device, terminal, server and methods for establishing a trustworthy connection between a user and a terminal

In Anspruch genommene Priorität(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat: State: Pays:	Tag: Date: Date:	Aktenzeichen: File no. Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:
/

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE
Etats contractants désignés lors du depôt:

Bemerkungen:
Remarks:
Remarques:

SZ 9-98-041 29 January 1999

Personal device, terminal, server and methods for establishing a trustworthy connection between a user and a terminal

The invention relates to situations where untrusted terminals are used to access a computer system. More particularly, it relates to public untrusted terminals which are connected via a network to a computer system and the authentication of such public untrusted terminals.

TECHNICAL FIELD AND BACKGROUND OF THE INVENTION

Automatic teller machines (ATM) and Internet kiosks are typical examples of public untrusted terminals which are used to access computer systems. A typical system is illustrated in Figure 1. A user 1 is considered, withdrawing money from an ATM 6 using a bank card 2. In all existing systems, users 1 have to enter a personal identification number (PIN) or pass-phrase in order to reliably authenticate themselves to the bank. But there is no way for the user 1 to authenticate the bank, respectively the ATM 6. There have been incidents where thieves set up fake ATMs and successfully stole PINs and magnetic stripe information from unsuspecting users.

The same fake-terminal problem occurs in many other settings as considered in the following.

ATMs and point-of-sale terminals: In both scenarios, every user 1 is registered with a specific server 5 (e.g., a credit-card issuer). All transactions of the user 1 are eventually authorized by the server 5. Servers 5 can typically identify and authenticate legal terminals 6. A typical attack scenario is when the attacker would set up an illegal terminal 6 which waits for the user 1 to type in the PIN code, read any necessary information from the card 2, and then refuse service, for example by displaying a "terminal out of order" message. Unsuspecting users 1 will simply move on to a different terminal 6. The attacker can later use the stolen information at a legal terminal 6.

Public Internet kiosks: Short-term access to the Internet from public terminals is an increasingly common feature in malls, airports, the so-called ``Internet cafés," and other public places. There is little risk for users who merely want to ``surf" the web from these terminals. But people can, and do, perform more security-sensitive transactions such as accessing their personal or business computer systems, making payments etc. from public Internet kiosks. This scenario differs from the previous ones in some respects:

SZ 9-98-041

- the user 1 may access several servers 5 from the same terminal 6, and
- the types of private information which needs to be protected may not be fixed, or even known a priori.

A similar scenario arises in the case of virtual mall kiosks. Virtual mall kiosks allow prospective customers to browse through and purchase the wares advertised by shop-keepers in the virtual mall. Functionally, this scenario is similar to public Internet kiosks.

In specific settings, such as ATMs that use biometrics instead of passwords to authenticate, the fake-terminal problem can be avoided. However, the general problem remains. A solution to this general problem must take into account different scenarios where the resources available to a user may be different: a user may have a trusted personal device with its own display or may have only a standard integrated chip card (e.g. a smartcard) with no display attached, or, in the simplest and most common case, may not have any personal trusted device at all.

The article "Trusting mobile user devices and security modules" in Computer, innovative technology for computer professionaly, Feb. 1997, IEEE Computer Society, pp 61-67 a simple protocol is described, where a user can authenticate a user device with display.

OBJECT AND ADVANTAGES OF THE INVENTION

It is an object of the present invention to provide a scheme to solve the problems associated with untrusted public terminals.

It is another object of the present invention to provide a scheme for a user to authenticate a public terminal before using it to process secutity-sensitive information.

These and other objects are accomplished by a device, terminal, server, communication system, and a method, as claimed in claims 1 - 41.

The personal device according to claim 1 offers the advantage that the user can authenticate an unknown and hence ex ante untrusted terminal and thereby find out whether the terminal can be trusted or not.

When the device comprises stored predetermined authentication information which is communicatable to the terminal, the device need not have an output means for outputting

the authenticity output message. The device hereby takes advantage of the output capability of the terminal and the trusted pathe that has been established before.

Using a third authentication step for the personal device to authenticate itself to the terminal brings in the advantage that not only the device can trust the server and via the server the terminal, but also the terminal can trust the device. Thus, also the terminal has the possibility to detect a fraudulent device and to interrupt secutity-sensitive applications upon detection of illegal personal devices.

Using bidirectional authentication steps is advantageous, since then bothe parties in the authentication upon success trust each other which results in a fully bidiractional trusted channel. Security-sensitive information can be exhchanged hence also bidirectionally. The third authentication step may then be renounced.

Requesting the user to authenticate himself again is advantageous in that also the device sees whether it can trust the user. Thus, also the device has the possibility to detect a fraudulent user and to interrupt secutity-sensitive applications upon detection of illegal users.

It prooves of great practical advantage when the authenticity output message comprises visible and/or audible and/or tactile information, because this is human-interface-readable information which renders recognition of a trusted terminal uncomplicated and fast.

If the authenticity output message comprises at least one value for lookup in a table stored in the terminal, the personal device needs less memory space since a simple reference to a place in the lookup table suffices to identify the correct authenticity output information.

A scenario, where the authenticity output message is communicatable to the terminal by the server, the authenticity output message preferably having been transmitted to the server by the user, refers to a situation when the personal device not even is writable by the user. This opens the invention to the field of prefabricated, not-amendable personal devices, such as preprogrammed or prewritten smartcards or magnetic cards.

Higher security can be achieved, when the authenticity output message is communicatable to the terminal by the server upon successful authentication of the device to the server, because the authenticity output message is safe in the server, as long as no authentication

has taken place. No attacker can hence somehow get the authenticity output message out of the device.

Using only part of the authenticity output message to be presented to the user, the achievable security is again higher, because the user can use the same authenticity output message several times without risking that an attacker somehow manages to spy out the output message and use it the next time to cheat on the user by using a fake terminal pretending to be a legal terminal.

SUMMARY OF THE INVENTION

The invention is related to a system which allows a user to autheticate unknown terminals. The user can hereby detect if a terminal he wants to use is a fake terminal or if it is a legal terminal and can be trusted. Only trusted terminals should be used to perform security-sensitive actions via the terminal. The invention uses a first authentication step wherein the terminal authenticates itself to a server. The authentication is either initiated simply by coupling the personal device to the terminal, or by some additional action performed by the user. The user can e.g. additionally press one or more buttons or keys on the terminal or on the personal device, wherever such input means are present. For authentication any known authentication system can be used, e.g. using a private-public key sytstem. Depending on whether the personal device has an own output means, such as a loudspeaker or a screen, the final message, whether the terminal can be trusted or not, can be output on the personal device or on the terminal itself. Since the user trusts his personal device, this message should come from the device itself. In the case, the device has no own output means, this message can hence be originating in the device and be from there transmitted to the terminal, which outputs it. The user can herefor input authenticaition information into his personal device which can then be fully or partially transmitted to the terminal. In the end, the terminal may use the transmitted information to give out the authenticity output message. After the first authentication step follows a second authentication step, wherein the server authenticates itself to the personal device, if there is one. Upon success of both authentication steps, the authenticity output message can be given to the user. If the personal device has no writing-capacity, the authentication information, also called authentication vector, can be transferred by the user via a trusted channel to the server. Upon successfull authentications, the server can then output some

message to the terminal to make it output the authenticity output message. The message from the server to the terminal can therefor be the authenticity output message itself, part of it, or any other kind of message that effects the issuance of the authenticity output message to the user. In the case, the user has no own personal device, also the method ca be used to transmit to the server the authentication vector before to approach a terminal. The user has agreed with the server on one or more tuples of challenge-response-authentication vector type. The authentication is performed via the challenge-response principle and upon successfull authentication, the server finally issues or has issued the authenticity output message via the terminal. The second messaging step, i.e. the output of the authenticity output message is preceded by a first messaging step which comprises the issuance of a message from the server. The message of the first messaging step tells that the terminal can be trusted.

In any of the embodiments, the messages that are transmitted, need not be transmitted in full. It may suffice to send only part of the message or some pointer to it and to have the final authenticity output message or the terminal authenticity message be looked up in a lookup table.

DESCRIPTION OF THE DRAWINGS

Examples of the invention are depicted in the drawings and described in detail below by way of example. It is shown in

Fig. 1 an arrangement with a device, a terminal and a server,

Fig. 2 a time scheme of a first method for establishing a trustworthy connection,

Fig. 3 a time scheme of a second method for establishing a trustworthy connection,

Fig. 4 a time scheme of a third method for establishing a trustworthy connection,

Fig. 5 a time scheme of a fourth method for establishing a trustworthy connection.

All the figures are for sake of clarity not shown in real dimensions, nor are the relations between the dimensions shown in a realistic scale.

SZ 9-98-041

DETAILED DESCRIPTION OF THE INVENTION

In the following, general scheme of the present invention and various exemplary embodiments thereof are described.

A typical system in which the present invention can be used is illustrated in Figure 1. The user 1 accesses a server system 5, hereinafter referred to as server 5, from a public untrusted terminal 6. This terminal 6 has a terminal output device 3, such as a screen or the like via which it communicates with the user 1. This terminal output device 3 has also means for the user 1 to communicate with the terminal 6, e.g. a keyboard. The terminal 6, respectively terminal output device 3 is connected to the server 5 via a network 4, which in its simplest form can be a direct line. For the purpose of accessing the server 5, the user 1 has an account on the server 5 which he trusts to correctly authenticate a public terminal 6. Public terminals are tamper-resistant but an attacker can easily replace a legal terminal 6 with a fake terminal or install a new fake terminal in a plausible location. The server 5 knows about legal terminals 6 and can authenticate them. Information necessary for a user 1 to authenticate the server 5 and, where necessary, information needed for the server 5 to authenticate the user 1, is set up during known user registration or other initialisation steps (e.g., agreeing on a shared key). Once an entity authenticates another, a confidential, authenticated channel is established as a result. In other words, an attacker cannot hijack an authenticated channel resulting from the authentication procedure. The symbols U, T, and S, are herein used to identify the user 1, the terminal 6, and the server 5, respectively. When the user 1 has a trusted personal device 2, e.g. a smartcard, a handheld phone or a magnetic card, it is denoted by D.

The authentication steps mentioned above are implemented using authentication protocols. There are various well-known authentication protocols for performing both one-way and two-way authentication such as Secure Sockets Layer (SSL), KryptoKnight, and Kerberos. Details of SSL are described by Alan O. Freier, Philip Kariton, and Paul C. Kocher in "The SSL protocol: Version 3.0.", Internet Draft, 1996. KryptoKnight is addressed by R. Bird, I. Gopal, A. Herzberg, P. Janson, S. Kutten, R. Molva, and M. Yung in "Systematic design of a family of attack-resistant authentication protocols", IEEE Journal on Selected Areas in Communications, Vol. 11, No. 5, pp. 679-693, June 1993, for example. Kerberos is

SZ 9-98-041

described by John T. Kohl and B. Clifford Neuman in "The Kerberos network authentication service (V5)", Internet Request for Comment RFC 1510, 1993.

The solutions herein proposed assume the use of a suitable authentication protocol, which can be one of the above-mentioned protocols, or any other protocol that serves a similar purpose.

The server 5 may be replicated, thereby avoiding it from becoming a bottleneck. All copies of the server 5 need to be aware of the up-to-date set of legal terminals 6 and the information necessary to authenticate them. There may also be several servers 5, each responsible for a separate domain. In this case, it is assumed that the necessary infrastructure, e.g. a public-key infrastructure, for the servers 5 to authenticate each other exists. In either case, the number of the terminals 6 is likely to be several orders of magnitude higher than the number of the servers 5.

Case 1: Personal device with built-in output capability

First is considered the scenario where the user 1 has a full-fledged trusted personal device 2 with its own output channel, such as a screen of a handheld phone. The terminal 6 cannot access the device output channel. Consequently, the user 1 can be sure that any information is communicated to him via this output channel does in fact originate from his trusted personal device 2. In other words, there is a trusted path c0 between the trusted personal device 2 and the user 1 (St1a). When the user 1 (U) walks up to an untrusted terminal 6 (T), he couples his device 2 (D) to the terminal 6 (T) by some means, e.g., infrared link, physical connection (St1b, St1c), and a communication is performed. The corresponding message flow is schematically illustrated in Figure 2.

First, a first authentication step A I is performed during which the terminal 6 authenticates itself to the server 5.

1. U → D: (St1d) The user U requests the device D to authenticate the terminal 6 (T) it is attached to, e.g., by clicking on a button on D's display.
2. D → T: (St 2) The device D requests T to authenticate itself to the server S.
3. T → S: (St3a) T runs a one-way authentication protocol to the server S. If this succeeds, the server S knows that it has an authenticated channel S-T to T. This authenticated channel S-T is established as a first authenticated trusted connection c1 (St3b). The server 5 hence trusts the terminal 6.

Then a second authentication step A II is performed making use of the first authenticated trusted connection c1.

4. S → D: (St4a) The server S runs a one-way authentication protocol to the device D via the first authenticated trusted connection c1. If this succeeds, the device D knows that it has an authenticated channel S-D to the server S, which is tunneled through S-T. This authenticated channel S-D is established as a second authenticated trusted connection c2 (St4b).

As next step a first messaging step M I follows. The terminal sends a session key 'key' to the server 5 (St4c).This key can then be used by the server S and the terminal T to exchange information. Since the server trusts the terminal it can accept the key and use

it. Using this session key enhances secutity since an attacker trying to spy on the exchanged information, respectively modify it inbetween has neither a chance to read the exchanged information nor to modify it without the modification being detected. Using a session key, i.e. a new key for every new session, which is the uninterrupted use of the described system in exactly one configuration, increases security again, since even a key once spied out by an attacker is useless for the next session.

5. S → D: The server S sends a message to the effect "T is authentic" via S-D. This message is a terminal authenticity message m_t which arrives at the device 2 via the terminal 6 (St5a). In addition, the server S here sends additional information, such as the session key 'key', or one-time certificates, that are used by the device D and T for a third authetication step A III. In this step, an authentication protocol is run between the device D and T (St5b) and upon success of the authentication a secure channel D-T is constructed between themselves (St5c). This authenticated channel D-T is established as a third authenticated trusted connection c3 (St5c).

6. D → U: Next follows a second messaging step M II during which the device D displays a message to the effect "T is authentic according to S" to the user U. This message is called the authenticity output message m_o. The appearing authenticity output message m_o tells the user U that he can trust the terminal 6.

7. D → U: In scenarios where the user U has to authenticate to the server S, it can be done in a separate phase following the above exchange. For that, during a fourth authentication step A IV the device 2 may request the user 1 to autheticate himself to the device 2 (St7).

8. U → D: The user 1 answers the request by entering a piece of information which is suited to authenticate the user 1 as legal user. This piece of information is e.g. a personal identity number PIN or a pass phrase (St8).

As mentioned before, there are various well-known authentication protocols that may be used for the one-way authentication flows above, as well as in the scenarios below. In step 3, T could run a two-way authentication protocol. This would foil an attacker masquerading as the server S. In scenarios where the user U has to authenticate to the server S, step 4 can also be carried out in form of a mutual authentication exchange, renouncing

steps 7 and 8. As long as the user U is not identified to T or the server S, the itinerary of the user U is kept confidential from T.

The scheme described above can be summarized as follows. The personal device 2 is equipped with means such that it can be coupled to the terminal 6. It furthermore comprises code which, when being executed in the device 2, performs a method for establishing a trustworthy connection between the user 1 and the terminal 6. This terminal 6 is connected to and authenticatable by at least one server 5 which is authenticatable by the device 2. If the device 2 is coupled to the terminal 6, which coupling may be performed by physical, optical or wire-bound or wireless means, the following steps are carried out.

- A first authentication step AI is initiated during which the terminal 6 authenticates itself to the server 5. Upon success of this initiation, a first authenticated trusted connection c1 is established between the server 5 and the terminal 6.

- Then, a second authentication step AII is initiated during which - via the established first authenticated trusted connection c1 - the server 5 authenticates itself to the device 2. Upon success of this authentication, a second authenticated trusted connection c2 is established between the server 5 and the device 2.

- Then, a terminal authenticity message (m_t) is received by the device 2 during a first messaging step MI. This message is received from the server 5 via the established second authenticated trusted connection c2 and confirms the established authenticity of the terminal 6.

- Then, during a second messaging step MII, an authenticity output message (m_o) is provide by the device 2 to the user 1. This is done via an output of the device 2 and/or via a terminal output 3 of the terminal 6.

The personal device 2 might comprise stored predetermined authentication information (vec) which can be sent to the terminal 6 for it to create the authenticity output message (m_o). The preferred case might be that the authenticity output message (m_o) is sent by the server 5 to the terminal 6. This authenticity output message (m_o) might comprise visible, audible, or tactile information, e.g., one or more of the following: background color, foreground color, background pattern, sound, letters, numbers. Likewise, the authenticity

output message (m_o) might comprise at least one value for lookup in a table which is stored in the terminal 6, for example. The authenticity output message (m_o) might also have been transmitted by the user 1 to the server 5. This is preferably done via a trusted communication connection cs. The authentication steps AI, AII, and AIII might be bidirectional.

In the above scenario the terminal 6 is able to authenticate itself to the server 5 during the first authentication step AI such that upon success the first authenticated trusted connection c1 is established between server 5 and the terminal 6. Furthermore, the terminal 6 facilitates the establishment of the second authenticated trusted connection c2 between the server 5 and the device 2. For certain implementations the terminal 6 might need a terminal output 3.

Furthermore, the terminal 6 might comprise a stored lookup table which is accessible via the authenticity output message (m_o).

The server 5 is connected to the terminal 6 via the network 4 or a link and is able to authenticate the terminal 6 during the first authentication step AI. After the first authentication step AI a first authenticated trusted connection c1 is established between the server 5 and the terminal 6. The server 5 is furthermore enabled to authenticate itself to the device 2 during the second authentication step AII such that the second authenticated trusted connection c2 is established. Then, the server 5 sends the terminal authenticity message (m_t) to the device 2 via the established second authenticated trusted connection c2, to confirm the established authenticity of the terminal 6.

Case 2: Personal smartcard without output capability

Now a scenario is considered where the user 1 is equipped with the device 2, such as an integrated circuit card (e.g. a smartcard), which has no output capability. One could try to use the same solution for this scenario as well. However, the problem arises in step 6 since the device D does not have its own display. Consequently, it does not have a trusted path to the user U. There may be devices with other types of trusted paths, - e.g., mobile phones could use a speech synthesizer to communicate the message to the user U -, in which case the above described solution could still be used. Standard smartcards, however, have no such output mechanism. Hence one needs to modify the solution.

Customizing security-critical windows is a well-known security measure against Trojan horse attacks. There have been various proposals. One is described by N. Asokan et al. in " Preliminary report on basic services, architecture and design", Technical report, SEMPER Consortium, 1996. This Technical report is a SEMPER Project deliverable which was submitted to the European Commission; See http://www.semper.org for related information. Another proposal was pubslied by J.D. Tygar and A. Whitten in "WWW electronic commerce and Java Trojan horses" in Second USENIX Workshop on Electronic Commerce, pages 243-250, Oakland, California, November 1996. Some variants have also been implemented, for example in the SEMPER Trusted INteractive Graphical User INterface (see www.semper.org), or the hieroglyphs in the login dialog-box of the Lotus Notes software. While it is an effective countermeasure against simple-minded Trojan horses, it is ineffective in a scenario where the Trojan horse has read and write access to the display. As soon as a personalized window is displayed to the user, the Trojan horse program can read the personalization information, construct a fake window with the same information on top of the legitimate personalized window.

Hereinafter the personalization idea is combined with authentication protocols to achieve an effective solution for the scenario currently under consideration. In the current threat model, legal terminals are tamper-resistant while illegal terminals will not be able to authenticate themselves to the server 5. By not revealing the personalization information before the terminal 6 has been authenticated, one can be safe even from sophisticated attacker programs. Herein, the stronger threat model is considered in which an attacker may subvert legal terminals by, for example, installing Trojan horses.

It is assumed that the user 1 has a trusted (home) base (such as a home PC) where he can prepare his device 2, e.g. a smartcard, before beginning his travel. For the preparation, the user 1 selects a predetermined authentication information vec, also called authentication vector. The authentication vector consists here of one or more types of authenticators. An authenticator of a particular type is such that

- it can take one of several values,
- each different value can be perceived by an unaided human and distinguished from other values.

Examples of types of authenticators are:

- arbitrary text phrase
- background colour (of the order 256 possible values)
- foreground colour (of the order 256 possible values)
- background pattern (of the order 16 different patterns)
- sound sequence (of the order of 256 different tunes)

Another example is to include text phrases that can be easily recognized by the user 1. A variety of means could be employed in order to show the words to the user 1: e.g., visual - by printing them on a screen -, aural. - by using a speech synthesizer-, or tactile, - by representing the words in braille -. Words and phrases constitute the most powerful type of authenticators since they can be drawn from a relatively large space, and they can be communicated to the user 1 in a variety of ways.

The steps performed for authenticating the untrusted terminal 6 to the user 1 are depicted in figure 3.

The trusted home base here constitutes the trusted path c0 (St1a) between the device 2 and the user 1. The user 1 hence trusts his device 2. To prepare for his travel, the user 1 performs a preparation step P I in that he picks one combination as the predetermined authentication information vec: for example, a tuple of the form *phrase = abracadabra, background-color = blue, foreground-color = white, background-pattern = grid, tune = jingle-bells* on his trusted home base and stores it on the smartcard 2 (St1b).

SZ 9-98-041

When the user 1 walks up to the untrusted terminal 6 and inserts his smartcard 1 into the terminal's reader (St1c, St1d), the following message flows take place:

1. U → T: In the first authentication step A I, the user U requests T to authenticate itself to the server S, e.g., by typing in the identifier of the server S and clicking on a button on T's display (St1e).

2. T → S: The terminal T runs a one-way authentication protocol to the server S (St2a). If this succeeds, the server S knows that it has an authenticated channel S-T to T. This authenticated channel S-T is established as the first authenticated trusted connection c1 (St2b). The server 5 hence trusts the terminal 6.

Then a second authentication step A II is performed making use of the first authenticated trusted connection c1.

3. S → D: The server S runs a one-way authentication protocol to the device D via the authenticated channel S-T (St3a). If this succeeds, the device D knows that it has an authenticated channel S-D to the server S which is tunneled through the authenticated channel S-T. This authenticated channel S-D is established as a second authenticated trusted connection c2 (St3b).

As next step a first messaging step M I follows. The terminal sends a session key 'key' to the server 5 (St3c).

4. S → D: The server S sends a message to the effect "T is authentic" via S-D. This message is the terminal authenticity message m_t which arrives at the device 2 via the terminal 6 (St4a). In addition, the server S here sends additional information, such as the session key 'key', or one-time certificates, that are used by the device D and T for a third authetication step A III. In this step, an authentication protocol is run between the device D and T (St4b) and upon success of the authentication a secure channel D-T is constructed between themselves. This authenticated channel D-T is established as a third authenticated trusted connection c3 (St4c).

5. D → T: Next follows the second messaging step M II during which the device D transmits a message to the effect "T is authentic according to S" to the user U. Since the device D has no own display it takes advantage of the display of the terminal 6. The

device D reveals the pre-selected authentication vector, respectively the predetermined authentication information vec to the terminal T (St5).

6. T → U: T shows the received authentication vector to the user U, respectively displays the predetermined authentication information vec or part of it on its terminal output device 3, e.g, by displaying the selected colours and background pattern, and playing the selected tune. This output information constitutes the authenticity output message m_o. The appearing authenticity output message m_o tells the user U that he can trust the terminal 6.

In other words, the device D reveals the authenticator to T only after the server S has certified that T is a legal terminal 6. The probability of an illegal terminal correctly guessing the authenticator of the user 1 is very small, e.g., of the order of one in 256 x 256 x 16 x 256 with the parameters suggested above. If a rogue terminal incorrectly guesses the authenticators of several users in close succession, it may be reported to responsible control authorities and thus be detected as an illegal terminal.

So far the user U is not identified to T or the server S. This helps to keep the itinerary of the user U confidential from T.

The following variations are possible:

Smartcards may not have sufficient memory to store an authenticator in its entirety. However, if the types of authenticators are pre-defined, the smartcard needs to store only an index and the terminal 6 can use the index to look up the authenticator in a table of all possible values for the different components.

SZ 9-98-041

Case 3: Non-writable personal smartcard without output capability

Some smartcards may not be writable by the user 1. In this case, the scheme is modified as shown in figure 4:

1. During the preparation phase P I, the user 1 selects the authentication vector and communicates it to the server 5 via a confidential, authenticated channel cS from his home base. As the authentication vector, respectively as the predetermined authentication information vec, the user 1 picks one combination: for example, a tuple of the form *phrase = abracadabra, background-color = blue, foreground-color = white, background-pattern = grid, tune = jingle-bells*. Furthermore, still the trusted path c0 (St1a) between the device 2 and the user 1 exists. The user 1 hence trusts his device 2. When the user 1 walks up to the untrusted terminal 6 and inserts his smartcard 1 into the terminal's reader (St1b, St1c), the following message flows take place:

2. D → T: In the first authentication step A I, the device D requests T to authenticate itself to the server S (St2). This request is automatically induced by the insertion of the device D.

3. T → S: The terminal T runs a one-way authentication protocol to the server S (St3a). If this succeeds, the server S knows that it has an authenticated channel S-T to T. This authenticated channel S-T is established as the first authenticated trusted connection c1 (St3b). The server 5 hence trusts the terminal 6.

   Then a second authentication step A II is performed making use of the first authenticated trusted connection c1.

4. S → D: The server S runs a one-way authentication protocol to the device D via the authenticated channel S-T (St4a). If this succeeds, the device D knows that it has an authenticated channel S-D to the server S which is tunneled through the authenticated channel S-T. This authenticated channel S-D is established as a second authenticated trusted connection c2 (St4b).

   As next step a first messaging step M I follows. The terminal sends a session key 'key' to the server 5 (St4c).

5. S → D: The server S sends a message to the effect "T is authentic" via S-D. This message is the terminal authenticity message m_t which arrives at the device 2 via the terminal 6 (St5a). In addition, the server S here sends additional information, such as the session key 'key', or one-time certificates, that are used by the device D and T for a third authetication step A III. In this step, an authentication protocol is run between the device D and T (St5b) and upon success of the authentication a secure channel D-T is constructed between themselves. This authenticated channel D-T is established as a third authenticated trusted connection c3 (St5c).

6. S → T: Next follows the second messaging step M II during which the server S transmits a message to the effect "T is authentic according to S" to the user U. Since the device D has no own display, the server S takes advantage of the display of the terminal 6. The device D reveals the pre-selected authentication vector, respectively the predetermined authentication information vec to the terminal T (St6). This output information constitutes the authenticity output message m_o.

7. T → U: T shows the received authenticity output message m_o to the user U, respectively displays the authenticity output message m_o or part of it on its terminal output device 3, e.g, by displaying the selected colours and background pattern, and playing the selected tune. The appearing authenticity output message m_o tells the user U that he can trust the terminal 6.

The authentication step in step 5 is here necessary because the server S must not reveal the authentication vector to an attacker who is using a legal terminal 6 but pretends to be the user 1. The same authentication vector could be used several times. The user 1 could also select a set of authentication vectors during the preparation phase P I. Another variation is where the user 1 challenges the terminal T to show a different component of the authentication vector each time. This will also help foil an attacker who watches of a legitimate user 1 and learns his authentication vector.

As in the previous examples, the terminal T could run a two-way authentication protocol with the server S (Step 2). This would foil an attacker masquerading as the server S.

Case 4: No personal device

Smartcards and other personal trusted devices may become commonplace in the near future. But to date, their use is still limited. Most users are armed only with simple pass-phrases (e.g., in the case of Internet access) or memory cards (e.g., in the case of credit/debit cards). In this section, the scenario is investigated in which the user 1 has no personal computing device 2 at all. The corresponding steps are depicted schematically in figure 5.

A solution for one way authentication called S/Key, is described by N. Haller in "The S/Key one-time password system", Symposium on Network and Distributed Systems Security, Catamaran Hotel, San Diego, California, February 1994. Internet Society. This document is incorporated in its entirety.

Using such an the S/Key system, a server issues a number of challenge/response pairs to a user during an initialisation stage. The user prints out the list of these pairs. The responses are essentially one-time passwords. In order to access the system, the user identifies himself and the server sends a challenge. The user then looks up the appropriate response from his printed list, sends it back to the server, and strikes off that pair from his list. It is proposed to use an S/Key-like system in both directions.

Before beginning his travel, in the preparation step P I, the server S sends a number of challenge/response pairs to the user 1 via a confidential, authenticated channel c0 to the user's home base and the user 1 selects a different authentication vector for each challenge and sends the authentication vector / challenge pairs back to the server S. The user 1 also prints out the entire list of <challenge,response, authentication vector> triples.

When the user 1 walks up to an untrusted terminal 6, the following message flows take place:

1. U → T: : In the first authentication step A I, the user U requests the terminal T to authenticate itself to the server S, e.g., by typing in the identifiers of the user U and the terminal T, and clicking a button (St1).

2. T → S: the terminal T runs a one-way authentication protocol to the server S. If this succeeds, the server S knows that it has an authenticated channel S-T to the terminal T.

This authenticated channel S-T is established as the first authenticated trusted connection c1 (St2b). The server 5 hence trusts the terminal 6.

Then a second authentication step A II is performed making use of the first authenticated trusted connection c1.

3. S → T: the server S sends one of the challenges, previously exchanged with the user U during the preparation phase P I, via S-T to the terminal T (St3).

4. T → U: The terminal T displays the challenge to the user U (St4).

5. U → T: The user U looks up the response corresponding to the challenge on his printout and types it in, provided it is not already struck off (St5).

6. T → S: The terminal T sends the response via S-T to the server S (St6).

7. S → T: If the response is valid, the server S looks up the authentication vector corresponding to the challenge, and sends it via S-T to the terminal T (St7).

8. T → U: The terminal T shows the received authentication vector to the user U (St8).

The user U can verify if this is indeed the authentication vector corresponding to the challenge, according to his printed sheet. If so, he can be confident that the terminal T is a legal terminal 6. the user U then strikes off the entry corresponding to the challenge from his printed list. If the authentication fails, the user U as well as the server S should still cross out the entry corresponding to that challenge and never use it again. As before the terminal T could run a two-way authentication protocol with the server S (step 2). This would foil an attacker masquerading as the server S.

Variations of this scheme are addressed below.

The user 1 may want to avoid carrying around a printed list. It can also be a security weakness: if the attacker manages to get hold of the printed list, he can fool the user 1 and/or the server 5. In this case, he can make do with a single authentication vector. Steps 3-6 are then dropped. In step 7, the server S sends the authentication vector to the terminal T without any further checks. This simplification is not secure against targeted attacks where the attacker obtains the authentication vectors of specific users, e.g., by interacting with a legal terminal, sets up a fake terminal 6, and waits for these users to come

in. But it is useful against untargeted attacks, i.e., setting up a fake terminal 6 without specific users in mind. If users change their authentication vectors regularly, large-scale targeted attacks are not feasible.

The authenticity message can in principle also have been transmitted to the user by the server.

A second variation is, as in the previous scenario, the user 1 can be allowed to challenge the terminal T to show a different component of the authentication vector each time: i.e., the user 1 specifies the type of the authentication vector as the challenge since it may help the user 1 remember the challenges. For example, it is easier for a user 1 to remember a color, a tune, and a word rather than to remember three colors.

A user 1 need not necessarily remember his entire authentication vector, but need only be able to recognize incorrect authentication vectors. One possibility to construct authentication vectors with high entropy is to arrange them by themes. For example, the user 1 could issue a challenge on the theme "car," and ask for specific attributes of his car. A car has several attributes which are easy to recognize.

The approach can be summarized as follows:

(a) a first authentication step AI is executed during which the terminal 6 authenticates itself to the server 5. Upon success of the first authentication, a first authenticated trusted connection c1 is established between the server 5 and the terminal 6.

(b) during a second authentication step AII a challenge is received from the server 5 and output to the user 1.

(c) Then, a response is received from the user 1 and transmitted to the server 5. During a first messaging step M I an authenticity output message (m_o) is received at the terminal 6.

(d) During a second messaging step MII the authenticity output message (m_o) is communicated at least partially to the user 1 via an output 3 of the terminal 6.

The above-described approaches depend on the level of computational resources available to the user. In most cases untrusted terminals can be authenticated and secure session

established between the user and some remote server system for the exchange and/or processing of sensitive information.

Those skilled in the art will recognize that many modifications and changes can be made to the particular embodiments described above without departing from the spirit and scope of the invention.

All the described and depicted embodiments can be combined in total or in part.

CLAIMS

1. Personal device (2) to be connected to a terminal (6) and being equipped with a computerized method for establishing a trustworthy connection between a user (1) via the device (2) and the terminal (6) which is connected to and authenticatable by at least one server (5) which is authenticatable by the device (2), whereby in the case the device (2) is coupled to the terminal (6),

a first authentication step (A I) is initiatable during which the terminal (6) authenticates itself to the server (5), upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

subsequently a second authentication step (A II) is initiatable during which via the established first authenticated trusted connection (c1) the server (5) authenticates itself to the device (2), upon success of which, between the server (5) and the device (2), a second authenticated trusted connection (c2) is established, and

subsequently at the device (2) during a first messaging step (M I), from the server (5) via the established second authenticated trusted connection (c2), a terminal authenticity message (m_t) is receivable confirming the established authenticity of the terminal (6),

and subsequently during a second messaging step (M II) an authenticity output message (m_o) is communicatable from the device (2) to the user (1) via a device output of the device (2) and/or via a terminal output (3) of the terminal (6).

2. Personal device, according to claim 1, characterized in that it comprises stored predetermined authentication information (vec) communicatable to the terminal (6) for the terminal (6) to create the authenticity output message (m_o).

3. Personal device, according to claim 1 or 2, characterized in that it is able to authenticate itself to the terminal (6) during a third authentication step (A III).

4. Personal device, according to one of claims 1 to 3, characterized in that the initiatable authentication steps (A I-III) are bidirectional.

5. Personal device, according to one of claims 1 to 4, characterized in that it is able to request the user (1) to authenticate himself and that it is able to receive user input (PIN) for user authentication.

6. Personal device, according to one of claims 1 to 5, characterized in that the authenticity output message (m_o) comprises visible and/or audible and/or tactile information, e.g. one or more of the following: background color, foreground color, background pattern, sound, letters, numbers.

7. Personal device, according to one of claims 1 to 6, characterized in that the authenticity output message (m_o) comprises at least one value for lookup in a table stored in the terminal (6).

8. Personal device, according to one of claims 1 to 7, characterized in that the authenticity output message (m_o) is communicatable to the terminal (6) by the server (5), the authenticity output message (m_o) preferably having been transmitted to the server (5) by the user (1), preferably via a trusted communication connection (cs).

9. Personal device, according to one of claims 1 to 8, characterized in that the authenticity output message (m_o) is communicatable to the terminal (6) by the server (5) upon successful authentication of the device (2) to the server (5).

10. Personal device, according to one of claims 1 to 9, characterized in that the authenticity output message (m_o) is outputable only partially by the terminal (6), preferably according to a preselection from the user (1).

11. Terminal being enabled to be coupled to a personal device (2) and being equipped for establishing a trustworthy connection to a user (1) via the device (2), the terminal (6) being connected to and authenticatable by a server (5), the device (2) being able to authenticate the server (5), wherein in the case the device (2) is brought into contact with the terminal (6),

the terminal (6) is able to authenticate itself to the server (5) during a first authentication step (A I), whereby upon success between the server (5) and the terminal (6) a first authenticated trusted connection (c1) is established,

whereby upon the server (5) having authenticated itself successfully to the device (2) during a second authentication step (A II) via the established first authenticated trusted connection (c1), leading to the establishment, between the server (5) and the device (2), of a second authenticated trusted connection (c2) and upon

the device (2) having received from the server (5) a terminal authenticity message (m_t), via the established second authenticated trusted connection (c2), confirming the established authenticity of the terminal (6),

it is effected that by the device (2) to the user (1) an authenticity output message (m_o) is communicatable via a device output of the device (2) and/or via a terminal output (3) of the terminal (6).

12. Terminal, according to claim 11, characterized in that the first authentication step (A I) is effectable by an action of the user (1).

13. Terminal, according to claim 11 or 12, characterized in that from the terminal (6) after the first authentication step (A I) a session key (key) is issuable and communicatable to the device (2) via the server (5).

14. Terminal, according to one of claims 11 to 13, characterized in that it comprises a stored lookup table which is accessible via the authenticity output message (m_o).

15. Terminal, according to one of claims 11 to 14, characterized in that the authenticity output message (m_o) is receivable from the server (5), the authenticity output message (m_o) preferably having been transmitted to the server (5) by the user (1), preferably via a trusted communication connection (cs).

16. Terminal, according to one of claims 11 to 15, characterized in that the authenticity output message (m_o) is receivable from the server (5) upon successful authentication of the device (2) to the server (5).

17. Terminal, according to one of claims 11 to 16, characterized in that it is able to output the authenticity output message (m_o) only partially, according to a preselection from the user (1).

18. Server connected to a terminal (6) which is enabled to be coupled to a personal device (2), being equipped for establishing a trustworthy connection between a user (1) and the terminal (6) via the device (2), and being authenticable by the device (2), wherein in the case the device (2) is coupled to the terminal (6),

the server (5) is able to authenticate the terminal (6) during a first authentication step (A I), whereby upon success, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established,

the server (5) is furthermore able to authenticate itself to the device (2) during a second authentication step (A II) via the established first authenticated trusted connection (c1), whereby upon success, between the server (5) and the device (2), a second authenticated trusted connection (c2) is established and

the server (5) is furthermore able to send a terminal authenticity message (m_t) to the device (2), via the established second authenticated trusted connection (c2), confirming the established authenticity of the terminal (6),

whereby it is effected that by the device (2) to the user (1) an authenticity output message (m_o) is communicatable via a device output of the device (2) and/or via a terminal output (3) of the terminal (6).

19. Server, according to claim 18, characterized in that from the terminal (6) after the first authentication step (A I) a session key (key) is receivable and communicatable to the device (2).

20. Server, according to claim 18 or 19, characterized in that the authenticity output message (m_o) is sendable to the terminal (6), the authenticity output message (m_o) preferably having been transmitted to the server (5) by the user (1), preferably via a trusted communication connection (cs).

21. Server, according to one of claims 18 to 20, characterized in that the authenticity output message (m_o) is sendable to the terminal (6) upon successful authentication of the device (2) to the server (5).

22. Method, executable by a personal device (2), for establishing a trustworthy connection between a user (1) via the personal device (2) and a terminal (6) which is connected to and authenticatable by at least one server (5) which is authenticatable by the device (2), whereby in the case the device (2) is coupled to the terminal (6),

a first authentication step (A I) is initiated during which the terminal (6) authenticates itself to the server (5), upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

subsequently, after a second authentication step (A II) during which via the established first authenticated trusted connection (c1) the server (5) has successfully authenticated itself to the device (2), between the server (5) and the device (2), a second authenticated trusted connection (c2) is established, and

subsequently, at the device (2) during a first messaging step (M I), from the server (5) via the established second authenticated trusted connection (c2), a terminal authenticity message (m_t) is received confirming the established authenticity of the terminal (6),

and subsequently during a second messaging step (M II) an authenticity output message (m_o) is communicated from the device (2) to the user (1) via a device output of the device (2) and/or via a terminal output (3) of the terminal (6).

23. Method, according to claim 22, characterized in that stored predetermined authentication information (vec) is communicated from the device (2) to the terminal (6), for creating there the authenticity output message (m_o).

24. Method, according to claim 22 or 23, characterized in that the device (2) authenticates itself to the terminal (6) during a third authentication step (A III).

25. Method, according to one of claims 22 to 24, characterized in that the device (2) requests the user (1) to authenticate himself, whereafter the device is waiting to receive user input (PIN) for user authentication.

26. Method, according to one of claims 22 to 25, characterized in that the device (2) outputs the authenticity output message (m_o) in form of a visible and/or audible and/or tactile information, e.g. one or more of the following: background color, foreground color, background pattern, sound, letters, numbers.

27. Method, according to one of claims 22 to 26, characterized in that the authenticity output message (m_o) is communicatable to the terminal (6) by the server (5) upon successful authentication of the device (2) to the server (5).

28. Method, according to one of claims 22 to 27, characterized in that the authenticity output message (m_o) is output only partially by the terminal (6), according to a preselection by the user (1).

29. Method, executable by a terminal (6), for establishing a trustworthy connection between a user (1) via a personal device (2) and the terminal (6) which is connected to and authenticatable by at least one server (5) which is authenticatable by the device (2), whereby in the case the device (2) is coupled to the terminal (6),

a first authentication step (A I) is initiated during which the terminal (6) authenticates itself to the server (5), upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established,

in preparation of a second authentication step (A II) during which via the established first authenticated trusted connection (c1) the server (5) may authenticate itself to the device (2), and upon success of this authentication step, between the server (5) and the device (2), a second authenticated trusted connection (c2) is establishable, after which at the device (2) during a first messaging step (M I), from the server (5) via the established second authenticated trusted connection (c2), a terminal authenticity message (m_t) is receivable confirming the established authenticity of the terminal (6), and subsequently during a second messaging step (M II) an authenticity output message (m_o) is communicatable from the device (2) to the user (1) via a device output of the device (2) and/or via a terminal output (3) of the terminal (6).

30. Method, according to claim 29, characterized in that the first authentication step (A I) is effectable through an action of the user (1).

31. Method, according to claim 29 or 30, characterized in that from the terminal (6) after the first authentication step (A I) a session key (key) is issued and communicated to the device (2) via the server (5).

32. Method, according to one of claims 29 to 31, characterized in that a lookup operation is performed using a stored lookup table which is accessible via the authenticity output message (m_o).

33. Method, executable by a server (5), for establishing a trustworthy connection between a user (1) via a personal device (2) and a terminal (6) which is connected to and authenticatable by at least the server (5) which is authenticatable by the device (2), whereby in the case the device (2) is coupled to the terminal (6),

the server (5) authenticates the terminal (6) during a first authentication step (A I), whereby upon success, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established,

the server (5) authenticates itself to the device (2) during a second authentication step (A II) via the established first authenticated trusted connection (c1), whereby upon success between the server (5) and the device (2) a second authenticated trusted connection (c2) is established and

the server (5) sends a terminal authenticity message (m_t) to the device (2), via the established second authenticated trusted connection (c2), confirming the established authenticity of the terminal (6),

whereby it is effected that by the device (2) to the user (1) an authenticity output message (m_o) is communicatable via a device output of the device (2) and/or via a terminal output (3) of the terminal (6), having received a predetermined authentication information (vec) stored on the device (2).

34. Method, according to claim 33, characterized in that from the terminal (6) after the first authentication step (A I) a session key (key) is received and communicated to the device (2).

35. Method, according to claim 33 or 34, characterized in that the authenticity output message (m_o) is sent to the terminal (6), the authenticity output message (m_o) preferably having been transmitted to the server (5) by the user (1), preferably via a trusted communication connection (cs).

SZ 9-98-041

36. Method, according to one of claims 33 to 35, characterized in that the authenticity output message (m_o) is is sent to the terminal (6) upon successful authentication of the device (2) to the server (5).

37. Method, executable by a terminal (6), for establishing a trustworthy connection between a user (1) and the terminal (6) which is connected to and authenticatable by at least one server (5), the user (1) and the server (5) having agreed on at least one information-tupel containing a challenge, a response and an authenticity output message (m_o),

whereby a first authentication step (A I) is executed during which the terminal (6) authenticates itself to the server (5), upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

during a second authentication step (A II) the challenge is received from the server (5) and output to the user (1), and subsequently the response is received from the user (1) and transmitted to the server (5), and

during a first messaging step (M I) from the server (5) an authenticity output message (m_o) is received at the terminal (6),

and subsequently during a second messaging step (M II) the authenticity output message (m_o) is communicated at least partially to the user (1) via a terminal output (3) of the terminal (6).

38. Method, according to claim 37, characterized in that the first authentication step (A I) is effectable by an action of the user (1).

39. Terminal being equipped for establishing a trustworthy connection to a user (1) and being connected to and authenticatable by a server (5),

the user (1) and the server (5) having agreed on at least one information-tupel containg a challenge, a response and an authenticity output message (m_o),

the terminal (6) being able to authenticate itself to the server (5) during a first authentication step (A I) upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

upon reception, during a second authentication step (A II), of the challenge from the server (5), being able to output the challenge to the user (1), and subsequently to receive the response from the user (1) and to transmit it to the server (5), and

during a first messaging step (M I) being able to receive from the server (5) at least part of the authenticity output message (m_o),

and subsequently during a second messaging step (M II) to communicate at least part of the authenticity output message (m_o) to the user (1) via a terminal output (3) of the terminal (6).

40. Method, executable by a server (5), for establishing a trustworthy connection to a user (1) and being connected to and able to authenticate a terminal (6),

the user (1) and the server (5) having agreed on at least one tupel containing a challenge, a response and an authenticity output message (m_o),

whereby a first authentication step (A I) is executed during which the terminal (6) authenticates itself to the server (5), upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

during a second authentication step (A II) the challenge is sent to the terminal (6) from where the response is received, and

during a first messaging step (M I) by the server (5) the authenticity output message (m_o) is sent at least partially to the terminal (6),

and subsequently during a second messaging step (M II) at least part of the authenticity output message (m_o) is communicatable to the user (1) via a terminal output of the terminal (6).

41. Server being equipped for establishing a trustworthy connection to a user (1) and being connected to and able to authenticate a terminal (6),

the user (1) and the server (5) having agreed on at least one tupel containg a challenge, a response and an authenticity output message (m_o),

the server (5) being able to authenticate the terminal (6) during a first authentication step (A I) upon success of which, between the server (5) and the terminal (6), a first authenticated trusted connection (c1) is established, and

being able to send during a second authentication step (A II) the challenge to the terminal (6), which is able to output the challenge to the user (1), and subsequently to receive the response from the user (1) and transmit it to the server (5), and

after having received the response, during a first messaging step (M I) being able to send at least part of the authenticity output message (m_o) to the terminal (6) which subsequently during a second messaging step (M II) is able to communicate the authenticity output message (m_o) to the user (1) via a terminal output (3) of the terminal (6).

SZ 9-98-041

ABSTRACT

A personal device is to be connected to a terminal and being equipped with a computerized method for establishing a trustworthy connection between a user via said device and the terminal which is connected to and authenticatable by at least one server which is authenticatable by the device. In the case the device is coupled to the terminal, a first authentication step is initiatable during which the terminal authenticates itself to the server, upon success of which, between the server and the terminal, a first authenticated trusted connection is established. Subsequently a second authentication step is initiatable during which via the established first authenticated trusted connection the server authenticates itself to the device, upon success of which, between the server and the device, a second authenticated trusted connection is established. Subsequently at the device during a first messaging step, from the server via the established second authenticated trusted connection, a terminal authenticity message is receivable confirming the established authenticity of the terminal. Subsequently during a second messaging step an authenticity output message is communicatable from the device to the user via a device output of the device and/or via a terminal output of the terminal.

SZ 9-98-041

1/5

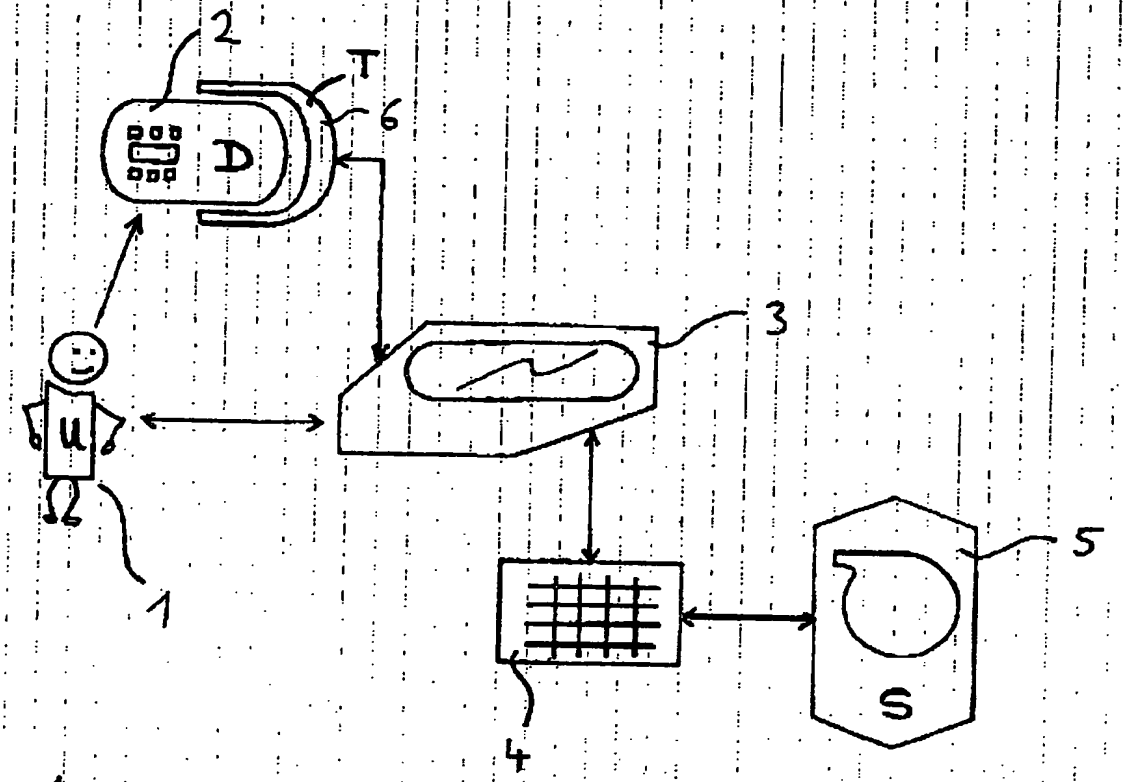

Fig. 1

2/5

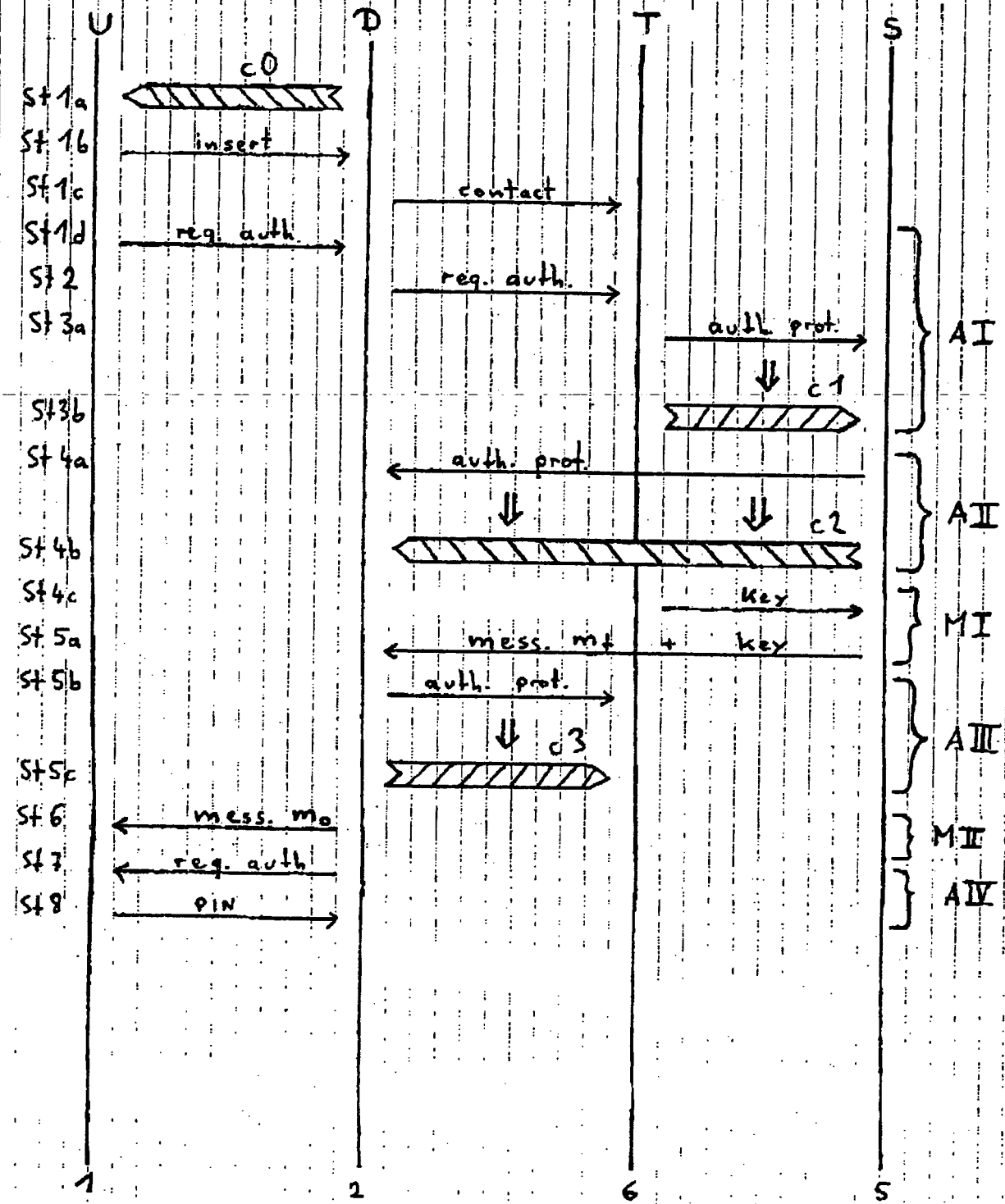
U
D
T
S
St 1a
c0
St 1b
insert
St 1c
contact
St 1d
req. auth
St 2
req. auth.
St 3a
auth. prot.
c1
St 3b
St 4a
auth. prot.
c2
St 4b
St 4c
key
St 5a
mess. m1 + key
St 5b
auth. prot.
c3
St 5c
St 6
mess. m0
St 7
req. auth
St 8
PIN
AI
AII
MI
AIII
MII
AIV
1
2
6
5

Fig. 2

>Z 9-98-041

3/5

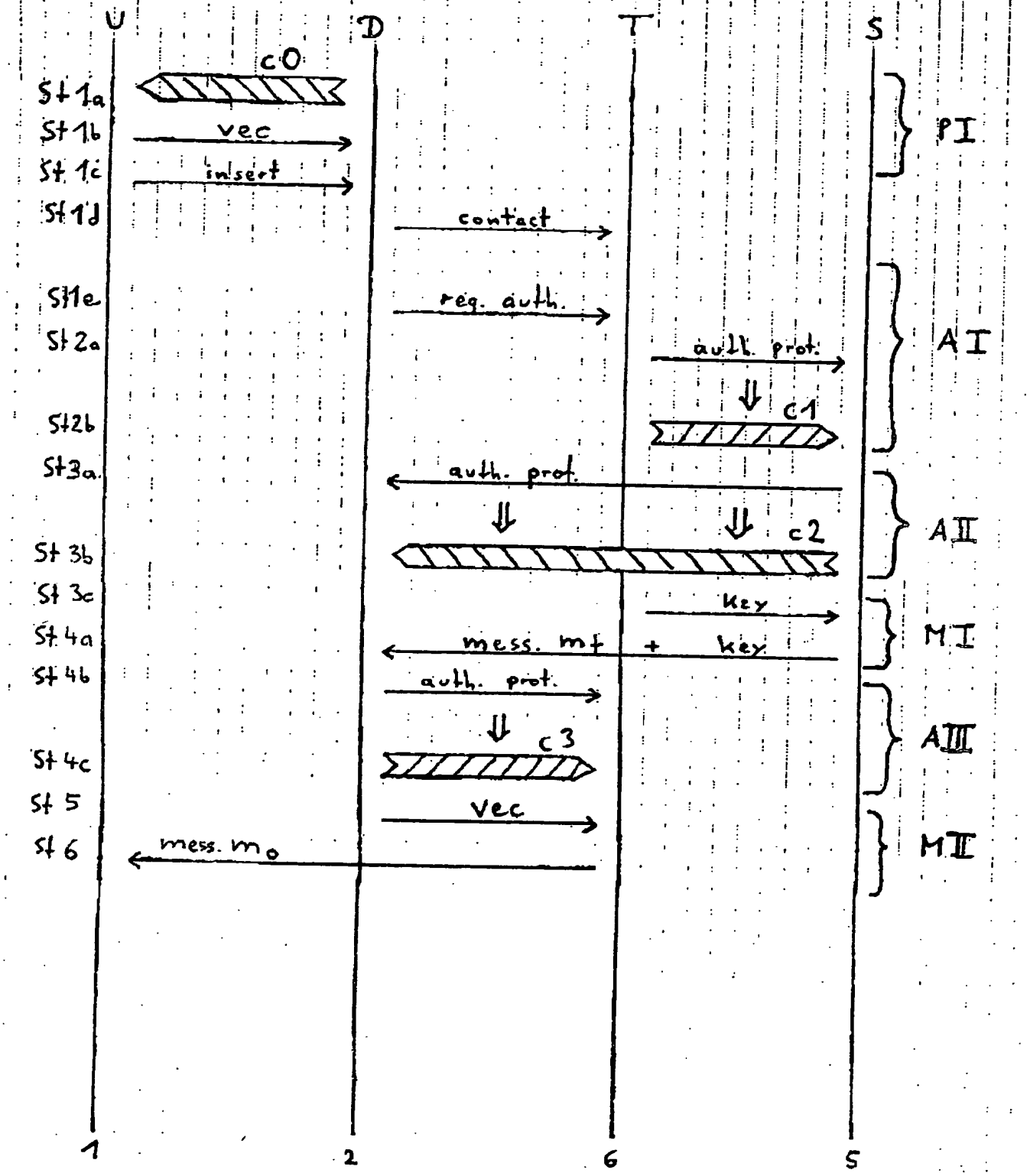

Fig. 3

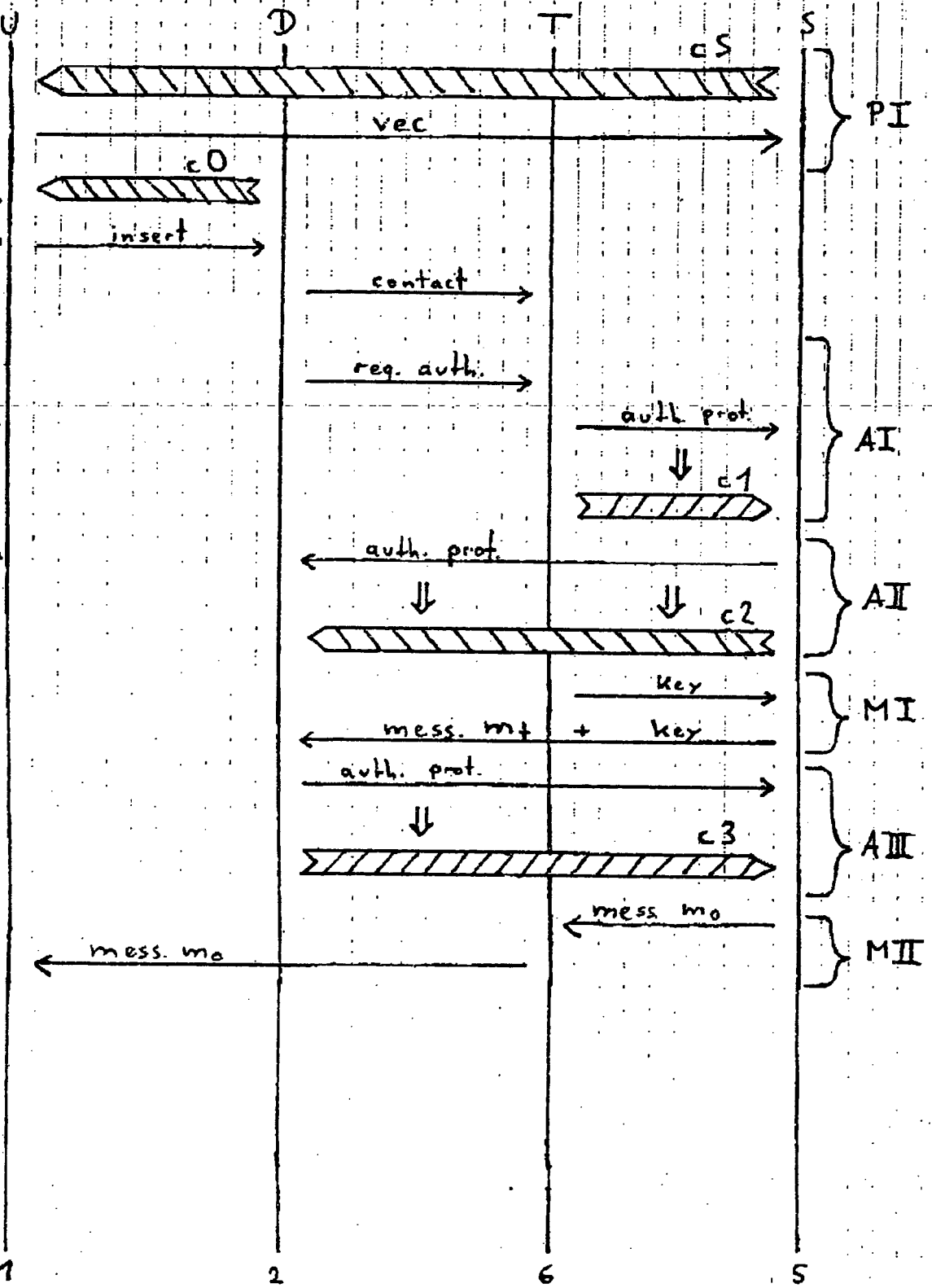
U
D
T
S
cS
PI
vec
c0
St 1a
St 1b
insert
St 1c
contact
St 2
req. auth.
St 3a
auth. prot.
AI
c1
St 3b
St 4a
auth. prot.
AII
c2
St 4b
St 4c
key
MI
St 5a
mess. mt + key
St 5b
auth. prot.
AIII
c3
St 5c
St 6
mess mo
MII
St 7
mess. mo
St 8a
St 8b
St 8c
St 9
1
2
6
5

Fig. 4

9-98-041

5/5

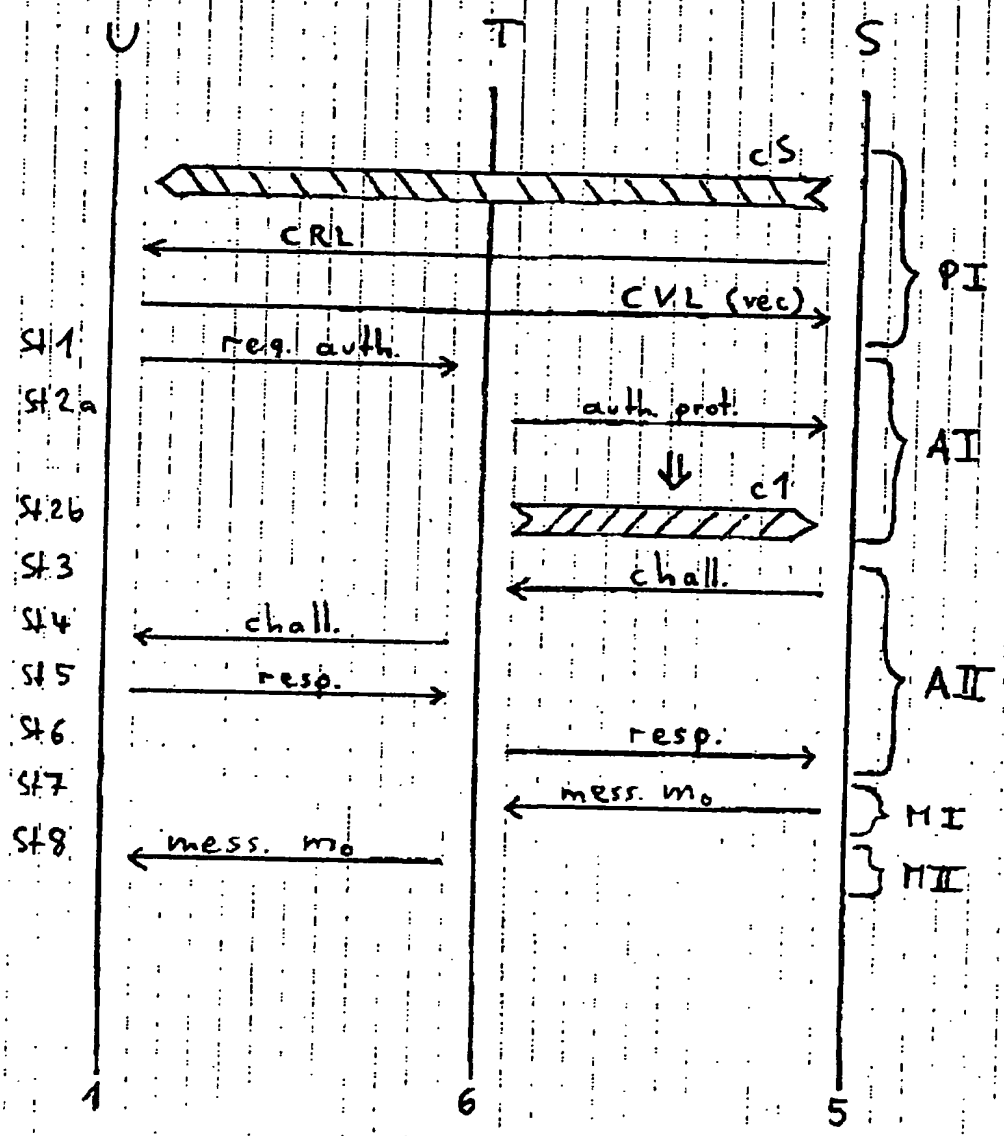

Fig. 5